

What Is Claimed Is:

1 1. An apparatus that translates host names into Internet Protocol (IP)
2 addresses, comprising:

3 a plurality of name servers, wherein each name server is configured to
4 translate a host name into a corresponding IP address; and

5 a plurality of load balancers coupled to the plurality of name servers,
6 wherein each load balancer is configured to,

7 receive requests for host name translations, and to
8 distribute the requests between the plurality of name servers
9 so as to balance load across the plurality of name servers;

10 wherein the plurality of load balancers are configured to operate in parallel
11 in distributing requests between the plurality of name servers.

1 2. The apparatus of claim 1, wherein each of the plurality of load
2 balancers is associated with its own IP address, and is configured to process
3 translation requests directed its own IP address.

1 3. The apparatus of claim 1, wherein each of the plurality of load
2 balancers is configured to take over load balancing operations for one or more
3 failed load balancers in the plurality of load balancers.

1 4. The apparatus of claim 3,
2 wherein load balancers in the plurality of load balancers are organized into
3 a ring; and

4 wherein each load balancer is configured to take over load balancing
5 operations for a neighboring load balancer in the ring, if the neighboring load
6 balancer fails.

1 5. The apparatus of claim 1, wherein each load balancer in the
2 plurality of load balancers is a proxy server that is configured to accept user
3 datagram protocol (UDP) and transmission control protocol (TCP) connections
4 from domain name system (DNS) clients, and to forward corresponding UDP or
5 proxy TCP requests to the plurality of name servers.

1 6. The apparatus of claim 1, wherein each of the plurality of load
2 balancers is configured to distribute translation requests between the plurality of
3 name servers based upon measured response times of the plurality of name
4 servers.

1 7. The apparatus of claim 1, further comprising an internal
2 communication network that couples the plurality of load balancers with the
3 plurality of name servers.

1 8. A method for translating a host name into an Internet Protocol (IP)
2 address, comprising:
3 receiving a translation request to translate the host name into the IP
4 address;
5 selecting a name server from a plurality of name servers to process the
6 translation request based upon a measured load of the plurality of name servers, so
7 that overloaded name servers will not be selected; and

8 forwarding the translation request to the selected name server so that the
9 selected name server can translate the host name into the IP address.

1 9. The method of claim 8, wherein receiving the translation request
2 involves receiving the translation request at one of a plurality of load balancers,
3 wherein each load balancer is configured to:

4 receive translation requests for host name translations; and to
5 distribute the translation requests between the plurality of name servers so
6 as to balance load across the plurality of name servers.

1 10. The method of claim 9, wherein each of the plurality of load
2 balancers is associated with its own IP address, and is configured to process
3 translation requests directed its own IP address.

1 11. The method of claim 9, further comprising taking over load
2 balancing operations, if necessary, for one or more failed load balancers in the
3 plurality of load balancers.

1 12. The method of claim 11, wherein the plurality of load balancers are
2 organized into a ring; and
3 wherein each load balancer is configured to take over load balancing
4 operations for a neighboring load balancer in the ring.

1 13. The method of claim 9, wherein each load balancer in the plurality
2 of load balancers is a proxy server that is configured to accept user datagram
3 protocol (UDP) and transmission control protocol (TCP) connections from

4 domain name system (DNS) clients, and to forward corresponding UDP or proxy
5 TCP requests to the plurality of name servers.

1 14. The method of claim 8, further comprising measuring a load on the
2 plurality of name servers by periodically:

3 sending an information request to each name server in the plurality of
4 name servers; and

5 measuring a response time to the information request for each name server
6 in the plurality of name servers.

1 15. A method for performing failovers between a plurality of load
2 balancers that are configured to balance requests for host name to IP address
3 translations between a plurality of name servers that are coupled to the plurality of
4 load balancers, comprising:

5 sending a keep alive packet to a first neighboring load balancer in the
6 plurality of load balancers;

7 waiting for a response to the keep alive packet in order to determine if the
8 first neighboring load balancer remains alive;

9 if the first neighboring load balancer does not remain alive, taking over
10 servicing of translation requests directed to the first neighboring load balancer.

1 16. The method of claim 15, further comprising:

2 receiving a second keep alive packet from a second neighboring load
3 balancer in the plurality of load balancers; and

4 sending a response to the second keep alive packet to the second
5 neighboring load balancer.

1 17. The method of claim 15, wherein each of the plurality of load
2 balancers is associated with its own IP address, and is configured to process
3 translation requests directed its own IP address.

1 18. The method of claim 15, wherein the plurality of load balancers are
2 organized into a ring; and

3 wherein each load balancer in the plurality of load balancers is configured
4 to take over load balancing operations for a neighboring load balancer in the ring.

1 19. The method of claim 15, wherein each load balancer in the
2 plurality of load balancers is a proxy server that is configured to accept user
3 datagram protocol (UDP) and transmission control protocol (TCP) connections
4 from domain name system (DNS) clients, and to forward corresponding UDP or
5 proxy TCP requests to the plurality of name servers.

1 20. The method of claim 15, further comprising distributing translation
2 requests between the plurality of name servers based upon measured response
3 times of the plurality of name servers.

1 21. An apparatus that translates host names into Internet Protocol (IP)
2 addresses, comprising:

3 a plurality of name servers, wherein each name server is configured to
4 translate a host name into a corresponding IP address; and

5 a plurality of load balancers coupled to the plurality of name servers,
6 wherein each load balancer is configured to,

7 receive requests for host name translations, and to

- 1 22. A method for translating a host name into an Internet Protocol (IP)
- 2 address, comprising:
 - 3 receiving a translation request at one of a plurality of load balancers to
 - 4 translate the host name into the IP address;
 - 5 selecting a name server from a plurality of name servers to process the
 - 6 translation request based upon a measured load of the plurality of name servers, so
 - 7 that overloaded name servers will not be selected;
 - 8 forwarding the translation request to the selected name server so that the
 - 9 selected name server can translate the host name into the IP address; and
 - 10 taking over load balancing operations, if necessary, for one or more failed
 - 11 load balancers in the plurality of load balancers;
 - 12 wherein each load balancer is configured to distribute the translation
 - 13 requests between the plurality of name servers so as to balance load across the
 - 14 plurality of name servers.

1 23. A method for performing failovers between a plurality of load
2 balancers that are configured to balance requests for host name to IP address
3 translations between a plurality of name servers that are coupled to the plurality of
4 load balancers, comprising:
5 distributing translation requests between the plurality of name servers
6 based upon measured response times of the plurality of name servers;
7 sending a keep alive packet to a first neighboring load balancer in the
8 plurality of load balancers;
9 waiting for a response to the keep alive packet in order to determine if the
10 first neighboring load balancer remains alive;
11 if the first neighboring load balancer does not remain alive, taking over
12 servicing of translation requests directed to the first neighboring load balancer;
13 receiving a second keep alive packet from a second neighboring load
14 balancer in the plurality of load balancers; and
15 sending a response to the second keep alive packet to the second
16 neighboring load balancer.